A Comparative Study of Hand-Gesture Recognition Devices for Games

Ahmed S. Khalaf[1(✉)], Sultan A. Alharthi[1], Ali Alshehri[2], Igor Dolgov[3], and Z. O. Toups[1]

[1] Play and Interactive Experiences for Learning Lab,
Computer Science Department, New Mexico State University, Las Cruces, USA
khalaf@nmsu.edu
[2] Electrical Engineering Department, New Mexico State University, Las Cruces, USA
[3] Psychology Department, New Mexico State University, Las Cruces, USA

Abstract. Gesture recognition devices provide a new means for natural human-computer interaction. However, when selecting these devices to be used in games, designers might find it challenging to decide which gesture recognition device will work best. In the present research, we compare three vision-based, hand-gesture devices: Leap Motion, Microsoft's Kinect, and Intel's RealSense. The comparison provides game designers with an understanding of the main factors to consider when selecting these devices and how to design games that use them. We developed a simple hand-gesture-based game to evaluate performance, cognitive demand, comfort, and player experience of using these gesture devices. We found that participants preferred and performed much better using Leap Motion and Kinect compared to using RealSense. Leap Motion also outperformed or was equivalent to Kinect. These findings were supported by players' accounts of their experiences using these gesture devices. Based on these findings, we discuss how such devices can be used by game designers and provide them with a set of design cautions that provide insights into the design of gesture-based games.

Keywords: Hand-gesture · Games · Game design · Leap Motion · Kinect · RealSense · Vision-based

1 Introduction

The proliferation of gesture-recognition technology enables players to interact with games in new ways [13,36,57]. Such devices enable players to use different parts of their body as a controller, increasing physical immersion and enabling designers to explore novel game mechanics and interaction methods [36,47].

Human hands are one of the most expressive body parts [14,40,43]. To detect the different hand movements and gestures, several technologies have been developed and employed for various domains and applications: education [60,62], entertainment [12,15,63], navigation [28], and training games [51,52]. However,

M. Kurosu (Ed.): HCII 2020, LNCS 12182, pp. 1–20, 2020.
https://doi.org/10.1007/978-3-030-49062-1_4

the accuracy and performance of these technologies vary greatly [7], and in some cases the characteristics of the technologies may be incompatible with the game design [8,32].

When selecting hand-gesture recognition devices, game designers might find it challenging to decide which device will work best for their game. The present research enables game designers to understand the differences between these devices, how to select them, and how to incorporate them within their games. Based on our motivation and prior research, we evaluate three commonly used hand-gesture recognition devices: Leap Motion (LM), Intel RealSense (RS), and Microsoft Kinect (MK). This comparison develops an understanding of the main factors that influence their performance and use in games. Based on our study of prior work [29,35,41,46,50,56,58], we hypothesized that LM would outperform MK and that MK would outperform RS in games across the following measures:

H1–H4: Game performance (H1: overall completion time, H2: completion time for small objects, H3: completion time for large objects, H4: error rate): LM $>$ MK $>$ RS.

H5–H10: NASA Task Load Index (NASA-TLX) [18,19] workload measurement item scores (H5: mental demand, H6: physical demand, H7: temporal demand, H8: effort, H9: frustration, H10: performance[1]): LM $<$ MK $<$ RS.

H11–12: Perceived comfort (H11) and accuracy (H12): LM $>$ MK $>$ RS.

To test our hypotheses, we developed a simple hand-gesture game, *Handy*. In this game, players move an embodiment [6] on the screen with the gesture device and collect gems by performing a “grab” hand gesture when the embodiment is over them. The goal of the game is to collect all the gems as fast as possible. The development of the game identified issues in developing cross-platform gesture-based systems. In a mixed methods, within-subjects study, 18 participants played *Handy* using the three different hand-gesture recognition devices: LM, MK, and RS. We investigated the impact of these devices on game completion time, error rate, cognitive workload, and player experience.

We found that players performed better and felt most comfortable when they used LM and MK. Players also reported that LM and MK are more accurate, and they felt most comfortable using these devices. Through this work, we contribute:

1. a comparative study of three different vision-based hand-gesture recognition devices and analyzed how they might impact performance, cognitive load, and player experience in games; and
2. provide a set of design *cautions* that game designer can take into consideration when working and designing with hand-gesture recognition devices.

In the reminder of this paper, we synthesize background on gesture-based interaction, prior hand-gesture studies, and gesture-based interaction in games. We then provide a description of the methodology, the design of the game *Handy*,

[1] In the NASA-TLX, the performance score is reverse-coded so that, like the other scores, a higher score is worse, thus, a high score on performance indicates worse perceived performance.

and the user study. Finally, we discuss our results and provide a set of design cautions for future hand-gesture based games.

2 Background

In this section, we synthesize prior research on gesture-based interaction, discuss prior hand-gesture comparative studies, and provide an overview of prior designs and studies of gesture-based interaction in games.

2.1 Gesture-Based Interaction

Gesture-recognition technology enables users to interact with systems and games with their body in novel ways [13,36,57]. For example, users can use their hands to create different mid-air gestures to perform a number of actions, such as scrolling through an application window, flipping a page in a digital book, or pointing and selecting a folder in a computer desktop [16,40]. Such natural human-computer interaction has being shown to be useful and effective in a number of domains, such as in education [60,62], entertainment [12,15,63], navigation [28], and training games [51,52].

To enable such gesture-based interaction, a number of technologies have been designed and developed that utilize different techniques and sensors. Based on prior work [1,2,11,43,57], these devices can be divided into three types: vision-based (e.g., Leap Motion [25]), sensor-based (e.g., Myo Armband [23]), and hybrid-based (e.g., Electronic Speaking Glove [2]).

Gestures can be divided into *dynamic* or *static* [16,43]. A dynamic gesture consists of the movement of the hand, arm, and/or fingers over time (e.g., waving, writing in the air), whereas a static gesture involves the hand remaining in a certain posture at a point in time (e.g., fist, open palm) [44]. Vision-based systems need three phases: *hand detection*, *hand tracking*, and *gesture recognition* [43]. Infrared (IR) sensing is one of the technologies that vision-based recognition devices use to detect gestures. Some IR sensors depend on an IR projector which projects different dots on the recognized object within the range of the device; then captures the different movements with a camera [54].

While these devices provide designers and users with a set of different affordance that can enable novel ways to interact with applications and games, they pose new challenges and limitations that are worth investigating. In this work, we focus on investigating the differences between three vision-based hand-gesture recognition devices in the context of games.

2.2 Prior Hand-Gesture Comparative Studies

Prior studies have addressed how gesture-based interaction compares to more traditional approaches [45,46], as well as investigated the performance of specific devices [9,58]. For example, Sambrooks and Wilkinson [45], conducted a comparative study between MK, mouse, and touchscreen, in which they concluded

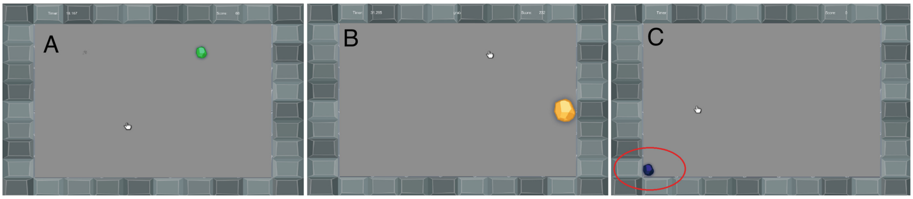

Fig. 1. The hand-gesture game *Handy*; the player collects gems of various sizes by moving their embodiment (a hand cursor) over the gem and executing a grab gesture. A: Collecting a small gem (green) near the top. B: Collecting a large gem (yellow) to the right. C: Collecting a small gem (black, circled in red) in the lower-right corner; in some cases, gems are placed in difficult areas that might require players to be more precise when moving their hand and making the grab gesture. (Color figure online)

that the performance of MK was much worse than mouse and touch screen for simple computer tasks. Furthermore, Seixas et al. [46] compared between LM, mouse, and touchpad, and found that LM performed poorly in pointing tasks compared to both mouse and touchpad. These studies together provide insights into the performance and usability of mid-air gesture device compared to traditional input modalities for 2D pointing tasks.

On the other hand, comparing between different mid-air hand gesture devices showed that these devices perform differently and can be selected based on the specific tasks [9,58]. Carvalho et al. [9], conducted a comparative study that considers LM and MK to evaluate their performance with relation to the user's age difference. Overall, device performance was consistent when comparing both devices side-by-side with specific groups of users; however, there was a significant difference in terms of performance of each device between some of the age groups. The authors reported that the performance of mid-air gesture devices could be varied based on age of the users. Furthermore, Weichert et al. [58] investigated the accuracy and robustness of LM. The authors conducted an experiment with an industrial robot to look at static gestures to drive the robot and dynamic ones to draw a path. The authors stated that the level of accuracy of LM is higher than other gesture devices within its price range, such as MK.

These studies together point to the advantages and disadvantages of these input modalities and provide insights into the performance of various gesture-recognition devices. However, they do not focus on vision-based gesture devices, do not provide enough insight into how such devices can be used in games, and are, thus, unable to assist game designers in making sound decisions about how to design and work with hand-gesture devices in games.

2.3 Gesture-Based Interaction in Games

Within the context of games, prior work investigated the performance and usability of different hand-gesture devices [30]. Correctly performing gestures in games can be part of the main mechanics, challenges, and experience of playing these games

[42,57]. However, hand-gesture recognition devices need to be responsive, consistent, intuitive, and comfortable for them to be used successfully in games [57].

Delays or lack of accuracy can negatively impact the experience of playing games. Moser and Tscheligi [35], examined the accuracy of LM when playing the physics-based game *Cut the Rope*, and found out that using gesture devices provided a positive player experience, however, a number of accuracy and orientation issues materialized when playing such games with a mid-air gesture device when compared to touch screens.

Muscle-controlled games, such as *The Falling of Momo* [51,52] have been developed to explore novel ways to utilize gesture recognition devices in physical therapy and training, and have been shown to be an effective mode of interaction.

Recognizing players hands is one of most common ways to enable interaction in virtual reality games [4,22,27,64]. Prior research investigated the use of such devices in VR games and how they influence players experience and the ability to perform different mid-air gestures correctly [24,26].

While prior research provides an understanding of how gesture recognition devices can be used in games, and sheds light on the challenges of using these devices, they do not provide game designers with an understanding of the differences between these gesture recognition devices and what they need to consider when designing games using this type of interaction.

3 Methodology

In this section, we explain the methods used to evaluate our hypotheses. We provide a detailed description of the process of recruiting participants, our hypotheses and experimental design, measures used, the game design, and study protocol.

3.1 Participants

We invited participants from New Mexico State University to volunteer for the formal study. Data collection occurred over a four-week period in the summer of 2018. 18 participants ($n = 18$, 3 female, 15 male; $M_{age} = 30.83$ years, $SD_{age} = 8.87$) were recruited. All participants were right-handed and were current computer science students.

3.2 Experimental Design

We used a within-subjects design with one independent variable (IV), device type, with three factors (i.e., LM, MK, RS). The dependent variables (DVs) were time taken to collect all objects, gesture recognition error rate, and workload. Game experience was taken as a co-variate (CV) (Table 1).

Participants played the game three times with each of the three devices. To rule out order effects, a complete counterbalance of the factors was achieved. With three conditions in this within-subject experiment, the total number of sequences required to achieve a complete counterbalance was six. We had a total of three participants in each sequence.

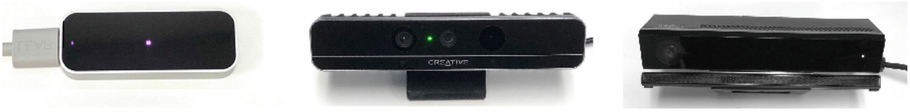

Fig. 2. Vision-based hand-gesture recognition devices compared in the present study. Left: Leap Motion (LM); middle: Intel RealSense (F200) (RS); right: Microsoft Kinect v2 (Windows Version) (MK).

Table 1. Description of the variables in this within-subject study.

Variable	Description
Hand-gesture devices (IV)	LM, RS, and MK
Game completion time (DV)	Time spent collecting all game objects in seconds
Error rate (DV)	Number of unrecognized hand-gestures
Cognitive workload (DV)	NASA-TLX used to assess workload
Game experience (CV)	Questionnaire used to assess game experience

Measures. Game completion time accounts for how long it took players to move from each gem to the next, performing the gesture accurately. We captured this data automatically with a logging system built into the game software, which separates how long it took the player to collect each goal.

Gesture error rate was determined by observing the player during gameplay. One researcher performed the observation and recorded false positives and negatives from the gesture system.

To assess cognitive workload, we used the NASA Task Load Index (NASA-TLX) [19], one of the most commonly used and the most widely validated of the various tools available for measuring workload [18]. NASA-TLX consists of six items measuring different aspects of workload (i.e., mental demand, physical demand, temporal demand, performance, effort, frustration) on a 100-point scale. Overall, higher values are worse, indicating that the participant found the task taxing or perceived low performance. In this study, the weighting component of NASA-TLX was omitted to reduce the time it took to complete the questionnaire [34].

To gauge subjective experience, after each session, we asked participants about the perceived accuracy of each device, perceived comfort, preference, and overall experience.

3.3 Research Artifact: *Handy*

To compare between the three hand-gesture devices, we designed *Handy*, a hand-gesture-based single-player game, in which the player uses their hand movements and gestures to play. The goal is to collect 36 objects that appear in a sequence and come in two different sizes (i.e., small, large) (Fig. 1). These objects need

Table 2. The hand gestures supported by each device.

Device	Supported hand gesture
LM	Different hand gestures can be defined using a detection utility [25]
RS	Open hand, close hand, grab, thumb up, thumb down, 'V' sign, two fingers pinch, full pinch, tap, wave, swipe, fingers spread
MK	Open hand, close hand, lasso

to be collected by placing the player's embodiment over the gem and using a "grab" gesture as fast as possible. The player's hand position is presented as a hand-shaped cursor embodiment on a 2D space. To successfully collect one object in the game, players need to perform the following:

- move their hand in any direction using hand movements with an open palm gesture and try to position their hand over the visible game object; then
- perform the "grab" gesture to successfully collect the game object.

On each play through, the player first has 6 randomly positioned targets, which do not count for score, followed by 30 that are pre-positioned. The pre-positioning was accomplished with an algorithm to randomly position the objects a consistent distance apart; the results were saved and used for each game round. This design enables the player to have a tutorial with the new device, followed by a consistent set of targets. Players are scored based on the time it takes them to collect each object. The gesture devices can be interchanged in the game for the purpose of this study.

3.4 Apparatus

Participants played *Handy* in a laboratory using a Windows PC (processor: Intel Core i7-5960X 3.00 GHz; RAM: 16 GB; graphics: NVIDIA GeForce GTX 980 Ti; monitor: 4K LED 27-in., wide screen), which was sufficient to run the game without performance issues. In each condition, the participants played the game with a different hand-gesture device. These vision-based hand-gesture recognition devices were selected based on their popularity and wide use within games:

- **Leap Motion**[2] (LM; Fig. 2, left): a small device for detecting hand position and gestures released in 2012. It is placed on a surface in front of a monitor, then can detect hands above it using IR sensors. The device contains two IR cameras and three IR LEDs [58]. We use LM SDK Orion Beta v3.2 to process captured data.
- **Intel RealSense**[3] (RS; Fig. 2, middle): a device for tracking human bodies, hands, and faces released in 2015. We use the F200 model, which uses an

[2] https://www.leapmotion.com.
[3] https://downloadcenter.intel.com/product/92255.

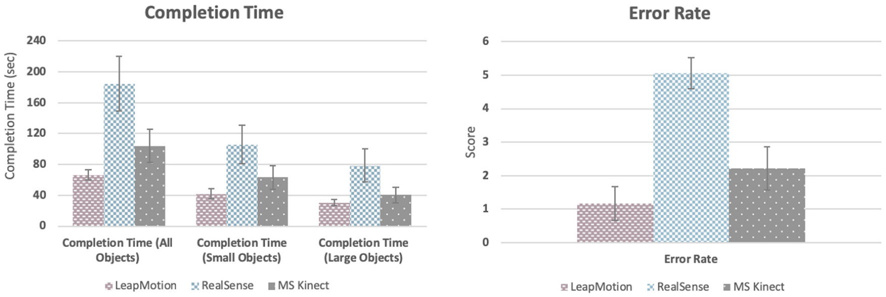

Fig. 3. Data plotted for completion time and error rate. Error bars represent 95% confidence intervals.

IR depth camera and RGB camera (1080p resolution) to capture imagery 20–120 cm from the device. We use the RS SDK version 2016 R2.

- **Microsoft Kinect**[4] (MK; Fig. 2, right): a motion-sensing input device by Microsoft for Xbox 360 and Xbox One video game consoles and Windows PCs, originally introduced in 2009, then upgraded in 2013 [55]. MK projects an IR dot pattern into the environment, which is uses for an IR depth camera; it also includes an RGB camera. For our study, we use the Kinect for Windows v2 and Kinect SDK v2.0.

3.5 Study Protocol

Before the beginning of the experiment, participants read and signed a consent form. Participants were then asked to complete the demographics questionnaire and asked about their prior experience with hand-gesture devices. Participants then played each of the three rounds of the game, with each of the different hand-gesture recognition devices. Between each round, they completed the NASA-TLX and the two Likert-scale-based questions. When participants finished the last session of the experiment, they were asked two open-ended questions to assess their overall experience.

4 Results

In this section, we present both the quantitative and qualitative results from our mixed methods within-subjects user study and discuss the main findings.

4.1 Performance

Repeated-measures analyses of variance (ANOVAs; IV: device) were used to evaluate the impact of using LM, RS, and MK on game performance, including completion time (overall, large objects, small objects) and error rate. The sphericity

[4] https://developer.microsoft.com/en-us/windows/kinect.

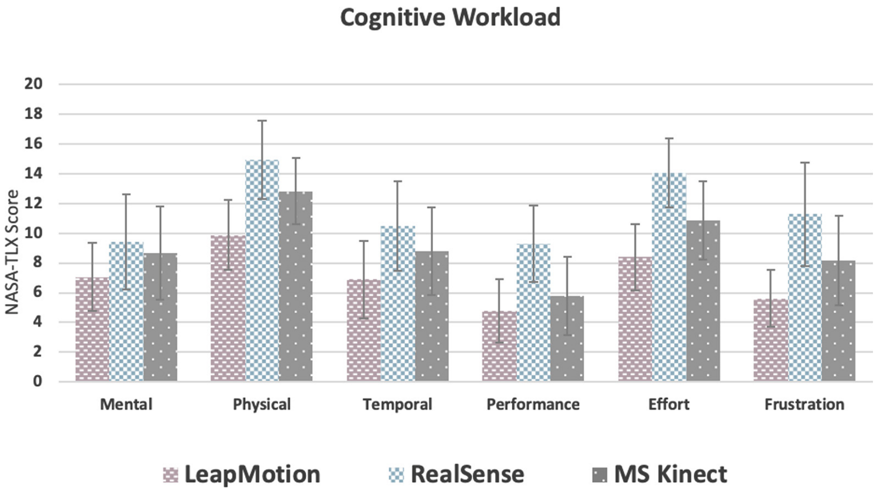

Fig. 4. Data plotted for cognitive workload. Error bars represent 95% confidence intervals.

assumption was violated for each of the completion time metrics (see Table 3), so we used the Hyunh-Feldt correction for degrees of freedom and significance in the corresponding ANOVAs.

As noted in Table 4 the main effect of device was significant across all of the behavioral measures; effect sizes were all very large. Pairwise comparisons showed RS to be worse than LM and MK across all measures ($p < 0.05$ in all cases). Pairwise comparisons also showed LM outperformed MK in measures of overall completion time, completion time for small objects, and error rate ($p < 0.05$ in all cases; see Fig. 3). So, hypotheses **H1**–**H4** were generally supported, except in instances where LM and MK performed similarly.

4.2 Cognitive Workload

Repeated-measures ANOVAs (IV: device) were used to evaluate the impact of using LM, RS, and MK on the measures included in the NASA-TLX. The sphericity assumption was only violated for temporal demand (see Table 3), so we used the Hyunh-Feldt correction for degrees of freedom and significance in the corresponding ANOVA.

As noted in Table 4, the main effect of device was significant across all of the TLX; effect sizes ranged from medium to small. Pairwise comparisons showed LM to be better than RS across all measures ($p < 0.05$ in all cases). Pairwise comparisons also showed MK to be better than RS across measures of perceived performance ($p < 0.05$) and marginally better ($p < 0.10$) in frustration. Additionally, pairwise comparisons showed LM to be similar to MK other than in temporal demand where LM performed better ($p < 0.05$) (Fig. 4). So, hypotheses **H5**–**H10** were generally supported, except in instances where self-reports of LM and MK use were similar.

Table 3. Mauchly's Test of Sphericity. When the sphericity assumption was violated ($p < 0.05$), we used the Hyunh-Feldt correction for degrees of freedom and significance in the corresponding analyses of variance.

Measure	Mauchly's W	DF	*p*-value	Epsilon (Hyunh-Feldt)
Completion time (all)	.460	2	.002	.681
Completion time (small)	.476	2	.003	.689
Completion time (large)	.460	2	.002	.681
Mental	.742	2	.092	.864
Physical	.898	2	.421	1.0
Temporal	.370	2	.0003	.637
Performance	.971	2	.790	1.0
Effort	.889	2	.391	1.0
Frustration	.994	2	.950	1.0
Comfort	.947	2	.646	1.0
Accuracy	.973	2	.802	1.0
Error Rate	.993	2	.944	1.0

Table 4. Results of the one-way ANOVAs for each of the DVs.

Measure	Treat. DF	Error DF	F value	*p*-value	η_p^2
Completion time (all)	1.361	23.144	28.915	<.0001	.630
Completion time (small)	1.379	23.439	14.862	.0003	.466
Completion time (large)	1.361	23.138	16.524	.0002	.493
Mental	2	34	1.689	.199	.09
Physical	2	34	6.154	.005	.266
Temporal	1.273	21.644	4.887	.030	.223
Performance	2	34	9.294	.0006	.353
Effort	2	34	7.462	.002	.305
Frustration	2	34	9.563	.0005	.360
Comfort	2	34	26.039	<.0001	.605
Accuracy	2	34	24.924	<.0001	.595
Error Rate	2	34	53.334	<.0001	.758

4.3 Perceived Comfort and Accuracy

We considered the responses to the questionnaire and coded the response values 1–5, with positive responses being higher. Repeated-measures ANOVAs (IV: device) were used to evaluate the impact of using LM, RS, and MK on measures of perceived comfort and accuracy. The sphericity assumption was not violated (see Table 3).

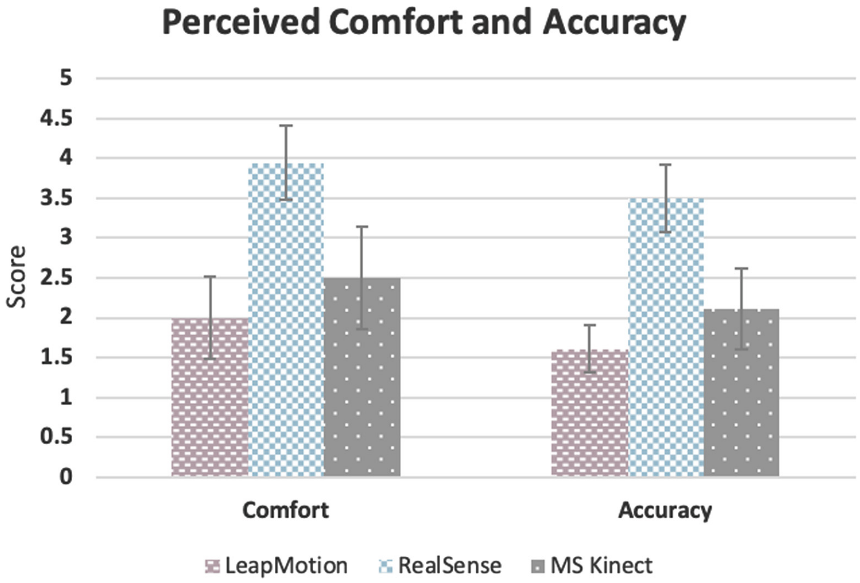

Fig. 5. Data plotted for perceived comfort and accuracy. Error bars represent 95% confidence intervals.

As noted in Table 4, the main effect of device was significant across both measures and the effect sizes were both large. Pairwise comparisons showed RS to be significantly worse than LM and MK across both measures ($p < 0.05$), whereas LM and MK did not differ from one another ($p < 0.05$) (Fig. 5). So, hypotheses **H11–H12** were partially supported.

4.4 Player Experience

We examined the participants' reflections on their experiences using these different hand-gesture devices in the game. Players stated that the accuracy of the three gestures devices varied:

> *I prefer Leap Motion because I could handle objects with better accuracy than Kinect which would be second choice.* [P15]

Players reported that they were more comfortable during the game when they used certain devices.

> *I was more comfortable when I used the first device [LM] to play the game.* [P17]

Some participants were more comfortable using MK:

> *I prefer Kinect because it was way more comfortable compared to other two devices.* [P13]

In contrast, players reported that they felt tired when they used RS to play the game.

> *I felt tired when I worked with the second device [RS], also, with it I lost the accuracy.* [P7]

We asked the participants about their prior experience with hand-gesture devices, and our data showed that participants had no prior experience with RS. Most of the participants have prior experience with LM and MK devices.

> *I liked to use Kinect to play the game because it is accurate and I am so familiar with it. I have been using it for three years now.* [P16]

5 Discussion and Design Cautions

In this section, we develop discussion and point out design *cautions* for designing and developing these types of games. Specifically, our take-away is that mid-air hand gesture devices are hard to design for and do not support the design of cross-platform games. Designers need to find novel solutions to overcome some of the challenges they may face when designing hand-gesture based games to provide players with an overall positive experience. The objective of this study was to compare three hand gesture recognition devices (LM, MK, RS) in the context of games and to provide insights and point to solutions for designing hand gesture-based games. Our results show that LM has better performance and accuracy compared to MK and RS, and players felt more comfortable when they played the game using LM and MK. These results align with prior research [49,58]. In the following, we present a number of cautions when designing and working with hand gesture-based devices. Attending to these cautions will help designers to improve their game design and player experience.

5.1 Arm and Body Fatigue

Our data showed that players felt tired when they played the hand-gesture game. We observed that some players experienced hand and arm fatigue. To overcome this, they switched to their left hand to support their right hand and to be able to continue playing the game. This problem is known as the "gorilla-arm effect" [20]. The reason for this problem is that participants had to raise and extend their arms and keep it in the air for extended periods while they were playing the game. This type of input modality is not suitable for long-term and extended use [5].

Designers of hand-gesture-based games need to understand the potential degrees of freedom for arms and hands [33,37,48], and be able to distinguish between which hand gestures are perceived as natural or easy to perform. Allowing players to use both hands to interact with the game can help players to switch between both hands during the game, which helps them rest and reduce any fatigue. An alternative solution is to position the gesture recognition device so that players do not raise their arms high. For example, a player could wear a small gesture recognition device, such as LM, and strap it to their thigh [28], allowing the arms to hang naturally at the sides of the body, reducing the need to raise their hands.

Another solution is to manipulate the level of interaction required by these games. For example, instead or requiring players to constantly interact with the game using their hands, the game can shift between different periods of active and regular hand-gesture interaction to periods of idling with essentially no interaction, allowing players to rest between these periods. This shift in interaction levels has shown to be effective in reducing *click fatigue* in clicker and idle games [3, 39].

We observed that some of these devices worked best while a player is standing instead of seating. Such differences in how these devices can be interacted with may cause not only arm fatigue, but also whole body fatigue. When selected these devices, designers need to consider whether the designed game is intended to be player while seating or standing to ensure that such design can be better supported by the device.

5.2 Challenges with Designing Hand-Gesture Games

Designing games that can be played using hand gestures poses challenges. Not all gesture-recognition devices support the same set of hand gestures. For example, RS can recognize multiple predefined hand gestures (e.g., thumbs up, two fingers pinch, full pinch), on the other hand, MK and Myo Armband do not support these gestures by default (see Table 2). Some of these devices enable designers to create their own set of hand gestures using SDKs, detection utilities, and toolkits, making it possible to build cross-platform and cross-device games. Designers need to make sure that their designs and gestures used are supported, this calls for a unified set of predefined gestures that devices need to support by default.

During the iterative process of designing our game *Handy*, we experimented with three different hand gestures (i.e., open hand, close hand, 'V' sign) that map to particular gems to be collected, which were color-coded. The problem with such a design is that it does not provide a natural mapping [38] between the action that the player is performing (i.e., collecting objects), and the hand gesture itself (e.g., 'V' sign) which could impact the performance of the players. Thus, avoiding unnatural gestures and mapping is critical to ensure the usability and comfort of such games. One solution is to focus on designing and supporting hand *microgestures* [10], or simple gestures that can be created in small interaction spaces. Further research is encouraged to provide game designers with a set of clearly identified natural mapped hand gestures and their possible uses within games (e.g., upward finger flick can be mapped to jumping) [48].

5.3 Size and Position of Game Objects

Our results showed that participants collected large game objects faster than small ones. We believe the reason for this is that using mid-air gesture devices is difficult when interacting with small objects. These results align with prior research [35], and are consistent with Fitts's Law [53]. Game designers should consider designing game objects that players and gesture devices can easily recognize and interact with.

Each vision-based gesture recognition device has a fixed detection range. For example, the detection range of RS for hand tracking and gesture recognition is 20–60 cm [21], if a player moves their hand closer to the camera, where the distance between the player hand and the device less than 20 cm (e.g., playing a 3D game), the device will stop detecting the player's hand. Thus, game designers need to take into consideration the detection range of these devices and make sure that the interaction space aligns with it.

5.4 Usability Considerations

Another caution is whether this kind of technology can be usable, responsive, intuitive, and comfortable within the context of games and how designer can find solutions for any usability problems. Our data indicated that participants were more accurate and perceived themselves to be more accurate when using some devices over others. We observed that such devices can easily lose track of the players' hands and fail to recognize the performed gestures. Due to such low accuracy level of most of these devices, designers need to make sure to provide appropriate feedback and awareness cues [17,59] to enable players to be aware if their hand gestures were correctly recognized or not. Objects that players need to interact with in the game should be made as obvious and accessible as possible. For example, avoiding the edges of the screen can help players maintain the position of their embodiment within the game.

While we only recruited right-handed players in this work, designers are encouraged to support both right and left-handed players to improve accessibility in games [61]. Our results indicated that participants preferred to use some devices over others for playing hand-gesture games, such preference was attributed to the accuracy, comfort, and familiarity with these devices. Designers of such games can make it possible for players to define or select their own gestures to play these games.

Finally, while providing players with the ability to interact with games using hand gestures can improve player experience and enjoyment [35], designers need to make sure that their games can still be played using other interaction modes if necessary.

6 Conclusions

In this study, we investigated three different vision-based gesture devices, LM, RS, and MK, in a computer game context to understand the differences in accuracy and performance between these devices. We found that participants preferred and performed better using the LM and MK than using the RS device. LM also outperformed MK along some metrics but not others, where they were equivalent. These findings were supported by players' accounts of their experiences using these gesture devices. Additionally, we found that participants preferred to used either LM or MK to play the game because of their improved accuracy (over RS) and prior familiarity with these devices.

While gesture devices proliferate and are capable of impressive hand tracking, they are still in a nascent stage. We found it challenging to develop similar experiences across multiple devices and ran into a number of issues with tracking errors in practice (even when playing in ideal conditions). We provide game designers with a set of design cautions and solutions to overcome some the challenges they might face when designing hand gesture games. It is our expectation that, as these technologies improve, these issues will get better.

7 Limitations

We acknowledge that this work is limited, and not intended to provide a complete analysis of the differences between hand-gesture devices for games. Here we focus on providing an initial understanding of how these devices might be different and how these differences might influence how they are used in games, which can help game designers to be aware of the challenges these hand-gesture devices might impose.

Our study was conducted with a small, specific sample (computer science students) which may limit the generalizability of the findings. Further research is encouraged to evaluate a wider range of devices with a larger sample of players using different off-the-shelf games in a more ecologically valid environment. We also did not conduct an analysis using Fitts's Law [31], doing that would have enabled us to conduct a more robust analysis, however, due to the design of the game and the collected data, such analysis is not possible. While the findings of this study provide insights and suggest a number of design cautions, they must be viewed within the limitations of the study.

References

1. Ahmed, M.A., Zaidan, B.B., Zaidan, A.A., Salih, M.M., Lakulu, M.M.B.: A review on systems-based sensory gloves for sign language recognition state of the art between 2007 and 2017. Sensors (Basel, Switzerland) **18**(7) (2018). http://europepmc.org/articles/PMC6069389
2. Ahmed, S.F., Ali, S.M.B., Qureshi, S.S.M.: Electronic speaking glove for speechless patients, a tongue to a dumb. In: 2010 IEEE Conference on Sustainable Utilization and Development in Engineering and Technology, pp. 56–60, November 2010
3. Alharthi, S.A., Alsaedi, O., Toups, Z.O., Tanenbaum, T.J., Hammer, J.: Playing to wait: a taxonomy of idle games. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI 2018, pp. 621:1–621:15. ACM, New York (2018). https://doi.org/10.1145/3173574.3174195
4. Alimanova, M., et al.: Gamification of hand rehabilitation process using virtual reality tools: using leap motion for hand rehabilitation. In: 2017 First IEEE International Conference on Robotic Computing (IRC), pp. 336–339, April 2017

5. Bachynskyi, M., Palmas, G., Oulasvirta, A., Steimle, J., Weinkauf, T.: Performance and ergonomics of touch surfaces: a comparative study using biomechanical simulation. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, pp. 1817–1826. ACM, New York (2015). https://doi.org/10.1145/2702123.2702607
6. Bayliss, P.: Beings in the game-world: characters, avatars, and players. In: Proceedings of the 4th Australasian Conference on Interactive Entertainment, IE 2007, pp. 4:1–4:6. RMIT University, Melbourne (2007). http://dl.acm.org/citation.cfm?id=1367956.1367960
7. Boulabiar, M.-I., Coppin, G., Poirier, F.: The issues of 3D hand gesture and posture recognition using the kinect. In: Kurosu, M. (ed.) HCI 2014. LNCS, vol. 8511, pp. 205–214. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07230-2_20
8. Cabreira, A.T., Hwang, F.: An analysis of mid-air gestures used across three platforms. In: Proceedings of the 2015 British HCI Conference, British HCI 2015, pp. 257–258. ACM, New York (2015). http://libezp.nmsu.edu:4009/10.1145/2783446.2783599
9. Carvalho, D., Bessa, M., Magalhães, L., Carrapatoso, E.: Performance evaluation of gesture-based interaction between different age groups using Fitts' law. In: Proceedings of the XVI International Conference on Human Computer Interaction, Interacción 2015, pp. 5:1–5:7. ACM, New York (2015). http://libezp.nmsu.edu:4009/10.1145/2829875.2829920
10. Chan, E., Seyed, T., Stuerzlinger, W., Yang, X.D., Maurer, F.: User elicitation on single-hand microgestures. In: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, CHI 2016, pp. 3403–3414. ACM, New York (2016). https://doi.org/10.1145/2858036.2858589
11. Cheng, J., Bian, W., Tao, D.: Locally regularized sliced inverse regression based 3d hand gesture recognition on a dance robot. Inf. Sci. **221**, 274–283 (2013)
12. Cook, H., Nguyen, Q.V., Simoff, S.: Enabling finger-gesture interaction with kinect. In: Proceedings of the 8th International Symposium on Visual Information Communication and Interaction, VINCI 2015, pp. 152–153. ACM, New York (2015). http://libezp.nmsu.edu:2763/10.1145/2801040.2801060
13. Doliotis, P., Stefan, A., McMurrough, C., Eckhard, D., Athitsos, V.: Comparing gesture recognition accuracy using color and depth information. In: Proceedings of the 4th International Conference on PErvasive Technologies Related to Assistive Environments, PETRA 2011, pp. 20:1–20:7. ACM, New York (2011). http://libezp.nmsu.edu:2763/10.1145/2141622.2141647
14. Erol, A., Bebis, G., Nicolescu, M., Boyle, R.D., Twombly, X.: Vision-based hand pose estimation: a review. Comput. Vis. Image Underst. **108**(1), 52–73 (2007). Special Issue on Vision for Human-Computer Interaction. http://www.sciencedirect.com/science/article/pii/S1077314206002281
15. Freeman, W.T., Ichi Tanaka, K., Ohta, J., Kyuma, K.: Computer vision for computer games. In: FG (1996)
16. Gardner, A., Duncan, C.A., Selmic, R., Kanno, J.: Real-time classification of dynamic hand gestures from marker-based position data. In: Proceedings of the Companion Publication of the 2013 International Conference on Intelligent User Interfaces Companion, IUI 2013 Companion, pp. 13–16. ACM, New York (2013). http://libezp.nmsu.edu:2763/10.1145/2451176.2451181
17. Gutwin, C., Greenberg, S.: A descriptive framework of workspace awareness for real-time groupware. Comput. Support. Coop. Work (CSCW) **11**(3), 411–446 (2002). https://doi.org/10.1023/A:1021271517844

18. Hart, S.G.: Nasa-task load index (NASA-TLX); 20 years later. Proc. Hum. Fact. Ergon. Soc. Annu. Meet. **50**(9), 904–908 (2006). https://doi.org/10.1177/154193120605000909
19. Hart, S.G., Staveland, L.E.: Development of NASA-TLX (task load index): results of empirical and theoretical research. Adv. Psychol. **52**, 139–183 (1988). Human Mental Workload. http://www.sciencedirect.com/science/article/pii/S0166411508623869
20. Hincapié-Ramos, J.D., Guo, X., Moghadasian, P., Irani, P.: Consumed endurance: a metric to quantify arm fatigue of mid-air interactions. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2014, pp. 1063–1072. ACM, New York (2014). http://libezp.nmsu.edu:2763/10.1145/2556288.2557130
21. Intel RealSense Technology: Intel RealSense, March 2016. https://software.intel.com/en-us/articles/intel-realsense-data-ranges
22. Khundam, C.: First person movement control with palm normal and hand gesture interaction in virtual reality. In: 2015 12th International Joint Conference on Computer Science and Software Engineering (JCSSE), pp. 325–330, July 2015
23. Labs, T.: Myo gesture control armband (2015). https://support.getmyo.com/hc/en-us/articles/203398347-Getting-started-with-your-Myo-armband
24. LaViola Jr., J.J.: Context aware 3D gesture recognition for games and virtual reality. In: ACM SIGGRAPH 2015 Courses, SIGGRAPH 2015, pp. 10:1–10:61. ACM, New York (2015). https://doi.org/10.1145/2776880.2792711
25. Leap Motion: Detection Utilities, March 2016. https://developer-archive.leapmotion.com/documentation/v2/unity/unity/Unity_DetectionUtilities.html
26. Lee, P.W., Wang, H.Y., Tung, Y.C., Lin, J.W., Valstar, A.: Transection: hand-based interaction for playing a game within a virtual reality game. In: Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems, CHI EA 2015, pp. 73–76. ACM, New York (2015). http://libezp.nmsu.edu:2763/10.1145/2702613.2728655
27. Lee, S., Park, K., Lee, J., Kim, K.: User study of VR basic controller and data glove as hand gesture inputs in VR games. In: 2017 International Symposium on Ubiquitous Virtual Reality (ISUVR), pp. 1–3, June 2017
28. Liu, M., Nancel, M., Vogel, D.: Gunslinger: subtle arms-down mid-air interaction. In: Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, UIST 2015, pp. 63–71. ACM, New York (2015). http://libezp.nmsu.edu:2763/10.1145/2807442.2807489
29. Lu, W., Tong, Z., Chu, J.: Dynamic hand gesture recognition with leap motion controller. IEEE Signal Process. Lett. **23**(9), 1188–1192 (2016)
30. Lv, Z., Halawani, A., Feng, S., Ur Réhman, S., Li, H.: Touch-less interactive augmented reality game on vision-based wearable device. Pers. Ubiquitous Comput. **19**(3–4), 551–567 (2015). https://doi.org/10.1007/s00779-015-0844-1
31. MacKenzie, I.S.: Fitts' law as a research and design tool in human-computer interaction. Hum. Comput. Interact. **7**(1), 91–139 (1992). https://doi.org/10.1207/s15327051hci0701_3
32. Matthies, D.J.C., Müller, F., Anthes, C., Kranzlmüller, D.: Shoesolesense: proof of concept for a wearable foot interface for virtual and real environments. In: Proceedings of the 19th ACM Symposium on Virtual Reality Software and Technology, VRST 2013, pp. 93–96. ACM, New York (2013). http://libezp.nmsu.edu:4009/10.1145/2503713.2503740
33. Morasso, P.: Spatial control of arm movements. Exp. Brain Res. **42**(2), 223–227 (1981). https://doi.org/10.1007/BF00236911

34. Moroney, W.F., Biers, D.W., Eggemeier, F.T., Mitchell, J.A.: A comparison of two scoring procedures with the NASA task load index in a simulated flight task. In: Proceedings of the IEEE 1992 National Aerospace and Electronics Conference, NAECON 1992, pp. 734–740. IEEE (1992)
35. Moser, C., Tscheligi, M.: Physics-based gaming: exploring touch vs. mid-air gesture input. In: Proceedings of the 14th International Conference on Interaction Design and Children, IDC 2015, pp. 291–294. ACM, New York (2015). http://libezp.nmsu.edu:4009/10.1145/2771839.2771899
36. Mueller, F.F., Byrne, R., Andres, J., Patibanda, R.: Experiencing the body as play. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI 2018, pp. 210:1–210:13. ACM, New York (2018). https://doi.org/10.1145/3173574.3173784
37. Napier, J., Napier, J.R., Tuttle, R.H.: Hands, vol. 9. Princeton University Press, Princeton (1993)
38. Norman, D.: The Design of Everyday Things: Revised and Expanded Edition. Basic Books, Revised 2013 edn. (2013)
39. Paavilainen, J., Hamari, J., Stenros, J., Kinnunen, J.: Social network games: players' perspectives. Simul. Gaming **44**(6), 794–820 (2013). https://doi.org/10.1177/1046878113514808
40. Peixoto, P., Carreira, J.: A natural hand gesture human computer interface using contour signatures (2005)
41. Pino, A., Tzemis, E., Ioannou, N., Kouroupetroglou, G.: Using kinect for 2D and 3D pointing tasks: performance evaluation. In: Kurosu, M. (ed.) HCI 2013. LNCS, vol. 8007, pp. 358–367. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39330-3_38
42. Rautaray, S.S., Agrawal, A.: Interaction with virtual game through hand gesture recognition. In: 2011 International Conference on Multimedia, Signal Processing and Communication Technologies, pp. 244–247, December 2011
43. Rautaray, S.S., Agrawal, A.: Vision based hand gesture recognition for human computer interaction: a survey. Artif. Intell. Rev. **43**(1), 1–54 (2015). https://doi.org/10.1007/s10462-012-9356-9
44. Reifinger, S., Wallhoff, F., Ablassmeier, M., Poitschke, T., Rigoll, G.: Static and dynamic hand-gesture recognition for augmented reality applications. In: Jacko, J.A. (ed.) HCI 2007. LNCS, vol. 4552, pp. 728–737. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73110-8_79
45. Sambrooks, L., Wilkinson, B.: Comparison of gestural, touch, and mouse interaction with Fitts' law. In: Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration, OzCHI 2013, pp. 119–122. ACM, New York (2013). http://libezp.nmsu.edu:4009/10.1145/2541016.2541066
46. Seixas, M.C.B., Cardoso, J.C.S., Dias, M.T.G.: The leap motion movement for 2D pointing tasks: characterisation and comparison to other devices. In: 2015 International Conference on Pervasive and Embedded Computing and Communication Systems (PECCS), pp. 15–24, February 2015
47. Spanogianopoulos, S., Sirlantzis, K., Mentzelopoulos, M., Protopsaltis, A.: Human computer interaction using gestures for mobile devices and serious games: a review. In: 2014 International Conference on Interactive Mobile Communication Technologies and Learning, IMCL 2014, pp. 310–314, November 2014

48. Sridhar, S., Feit, A.M., Theobalt, C., Oulasvirta, A.: Investigating the dexterity of multi-finger input for mid-air text entry. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, pp. 3643–3652. ACM, New York (2015). https://doi.org/10.1145/2702123.2702136
49. Staretu, I., Moldovan, C.: Leap motion device used to control a real anthropomorphic gripper. Int. J. Adv. Rob. Syst. **13**(3), 113 (2016). https://doi.org/10.5772/63973
50. Svoboda, J., Bronstein, M.M., Drahansky, M.: Contactless biometric hand geometry recognition using a low-cost 3D camera. In: 2015 International Conference on Biometrics (ICB), pp. 452–457, May 2015
51. Tabor, A., Bateman, S., Scheme, E., Flatla, D.R., Gerling, K.: Designing game-based myoelectric prosthesis training. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, CHI 2017, pp. 1352–1363. ACM, New York (2017). https://doi.org/10.1145/3025453.3025676
52. Tabor, A., Kienzle, A., Smith, C., Watson, A., Wuertz, J., Hanna, D.: The falling of momo: a myo-electric controlled game to support research in prosthesis training. In: Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts, CHI PLAY Companion 2016, pp. 71–77. ACM, New York (2016). https://doi.org/10.1145/2968120.2971806
53. Tognazzini, B.: First principles of interaction design. Interaction design solutions for the real world, AskTog (2003)
54. Ververidis, D., Karavarsamis, S., Nikolopoulos, S., Kompatsiaris, I.: Pottery gestures style comparison by exploiting myo sensor and forearm anatomy. In: Proceedings of the 3rd International Symposium on Movement and Computing, MOCO 2016, pp. 3:1–3:8. ACM, New York (2016). http://libezp.nmsu.edu:4009/10.1145/2948910.2948924
55. Vokorokos, L., Mihal'ov, J., Chovancová, E.: Motion sensors: gesticulation efficiency across multiple platforms. In: 2016 IEEE 20th Jubilee International Conference on Intelligent Engineering Systems (INES), pp. 293–298, June 2016
56. Vrellis, I., Moutsioulis, A., Mikropoulos, T.A.: Primary school students' attitude towards gesture based interaction: a comparison between microsoft kinect and mouse. In: 2014 IEEE 14th International Conference on Advanced Learning Technologies, pp. 678–682, July 2014
57. Wachs, J.P., Kölsch, M., Stern, H., Edan, Y.: Vision-based hand-gesture applications. Commun. ACM **54**(2), 60–71 (2011). http://libezp.nmsu.edu:2763/10.1145/1897816.1897838
58. Weichert, F., Bachmann, D., Rudak, B., Fisseler, D.: Analysis of the accuracy and robustness of the leap motion controller. Sensors **13**(5), 6380–6393 (2013). http://www.mdpi.com/1424-8220/13/5/6380
59. Wuertz, J., et al.: A design framework for awareness cues in distributed multiplayer games. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI 2018, pp. 243:1–243:14. ACM, New York (2018). https://doi.org/10.1145/3173574.3173817
60. Yao, Y., Chiu, P.T., Fu, W.T.: A gestural interface for practicing children's spatial skills. In: Proceedings of the 22nd International Conference on Intelligent User Interfaces Companion, IUI 2017 Companion, pp. 43–47. ACM, New York (2017). http://libezp.nmsu.edu:2763/10.1145/3030024.3038265
61. Yuan, B., Folmer, E., Harris, F.C.: Game accessibility: a survey. Univers. Access Inform. Soc. **10**(1), 81–100 (2011). https://doi.org/10.1007/s10209-010-0189-5

62. Zhang, K., Zhai, Y., Leong, H.W., Wang, S.: An interaction educational computer game framework using hand gesture recognition. In: Proceedings of the 4th International Conference on Internet Multimedia Computing and Service, ICIMCS 2012, pp. 219–222. ACM, New York (2012). http://libezp.nmsu.edu:2763/10.1145/2382336.2382398
63. Zhang, X., Chen, X., Wang, W.H., Yang, J.H., Lantz, V., Wang, K.Q.: Hand gesture recognition and virtual game control based on 3D accelerometer and EMG sensors. In: Proceedings of the 14th International Conference on Intelligent User Interfaces, IUI 2009, pp. 401–406. ACM, New York (2009). http://libezp.nmsu.edu:2763/10.1145/1502650.1502708
64. Zhu, Y., Yuan, B.: Real-time hand gesture recognition with kinect for playing racing video games. In: 2014 International Joint Conference on Neural Networks (IJCNN), pp. 3240–3246, July 2014

Author Queries

Chapter 4

Query Refs.	Details Required	Author's response
AQ1	This is to inform you that corresponding author and email address have been identified as per the information available in the Copyright form.	
AQ2	Per Springer style, both city and country names must be present in the affiliations. Accordingly, we have inserted the city and country names "Las Cruces and USA" in affiliations "1, 2 and 3". Please check and confirm if the inserted city and country names are correct. If not, please provide us with the correct city and country names.	